**Retrocomputing** Beta

# Were there ever plans published, or kits, or a machine made, for a noughts-and-crosses computer game which did not use electronics? [closed]

Asked 9 days ago Active yesterday Viewed 432 times

-3

**Closed.** This question is off-topic. It is not currently accepting answers.

**Want to improve this question?** Update the question so it's on-topic for Retrocomputing Stack Exchange.

Closed 5 days ago.

Improve this question

Were there ever plans published, or kits, or a machine made, for a noughts-and-crosses computer game which did not use electronics – using only wire, an electricity source, and light bulbs ?

- . A Playing against the computer game .
- . I'm looking for the most optimal solid-state designs
- . Machines containing things like electromagnetic relays / solenoids, are not accepted as answers, since I'm looking for alternate computer logic .
- . Machines containing multi-contact-switches would qualify, although, of course I'm also looking for the most optimal solid-state designs, which would decrease these to minimal amounts, perhaps at the cost of containing more wiring.
- . Machines containing a rotating component, like How to build a working digital computer, are not preferred answers, but they would be interesting to know about, if they only use wire, an

electricity source, and light bulbs ?

---

Why I think one may have existed, and how it could have functioned .

- It would use memory-boards, to compare the situation on the playing-board, to match it to one of the memory-boards, so when a memory-board is matched, it lights up a new square on the playing-board as the computer's next move.
- Each memory-board would just be a length of wire, with gaps at certain positions to represent noughts or crosses (it may require separate memory-boards, one for noughts, one for crosses), and both sides of the gap wired up to a square on the playing-board, so when the human plugs in a piece, it closes/completes that part of the wire to allow electricity to flow. (so each square on the playing-board, would be wired up to lots of memory-boards).
  - Maybe, to significantly reduce the amount of memory-boards, some sort of board rotation for matching), could be used, simply using wiring.
  - **NOTE** - An innovation ( a computing / logic / circuit innovation ) would be, that for the human player , each plug-in-piece, would be numbered from 1st move to last move, each one of these plug-in-pieces would be of a different length to reach down to activate a different move-number/platform/circuit of the game.
  ( Or, of course, the plug-in-piece could have multi-contact-switches, varying for the move number ).
- I wonder if theories used in Multiway switching / Switching circuit theory may be helpful. - I don't know if such a machine could work, I have not fully thought through all the details.
  - Anyone trying to visualize if this would work, use a 2x2 board.
- (Note – for a computer, where a person simply plays against another person, it would be very simple for a machine not using any electronics, just using wire, electricity-source, and light-bulbs, to simply detect a win, when an entire row has noughts or crosses. I assume the original computer battleships game worked like that, so that's not the question in this post.)
- Also, my other question, which they erased / closed Could you make a calculator without electronics ( using only, wire, electricity-source, and light-bulbs )? , had a useful innovation regarding the 'carry' in a binary calculation, it could be relevant for this question also.
  ( Quote "An innovation, could be plug-in-pieces/electrical-contact-pieces ( acting like switches ) that would be of a different length ( depending on what part of a number they are ) to reach down to activate a different circuit of the calculation ( useful for the 'carry' of a binary calculation ). ( Or, of course, the plug-in-piece could have multi-contact-switches, depending on what part of a number they are )."
  - **NOTE**. - What if this innovation, detailed above, regarding the 'carry' of a binary calculation, could lead to making an arithmetic-logic-unit which contains no transistors ( or even to other integrated-circuits not containing transistors ), by erasing that question there is no proof that I may have invented it.

hardware

Share Improve this question Follow edited yesterday asked Mar 14 at 2:36

mnml
405 2 6

Comments are not for extended discussion; this conversation has been moved to chat. – Chenmunka ♦ Mar 16 at 16:18

3 There are attempts to re-open this question using trivial edits. (Also its duplicate retrocomputing.stackexchange.com/q/18293/10260) – DrSheldon yesterday

## 3 Answers

Active | Oldest | Votes

6

Yes.

In 1978 Danny Hillis and Brian Silverman, two sophomores at MIT, built an entirely mechanical tic-tac-toe (noughts-and-crosses) playing machine. It contains absolutely no electronics.

This design could easily be converted to a device that uses wires and electric lights instead of little wooden flags to display its results.

It currently resides in the Computer History Museum in Mountain View, CA.

Share Improve this answer Follow answered Mar 14 at 4:05

A. I. Breveleri
722 3 8

Good reference, but I definitely saw a switches-and-lights implementation when I was aged 11 or so, which was quite a bit before 1978. I'm pretty sure that the builder did not invent it, but he's dead so I cannot ask, – another-dave Mar 14 at 4:10

1 The description at the link is hilarious! – snips-n-snails Mar 14 at 17:14

I wonder if that's the same machine or a different one from the one I saw at the computer museum

in/near Boston, Massachusetts some decades ago? – supercat Mar 14 at 19:21

@A.I.Breveleri: Scientific American showed an alternative version (still using Tinker Toy brand materials) which was smaller and I think more interesting. The one I saw at the Computer Museum used lots of vanes and purely combinatorial logic, but the one in Scientific American used sequential logic, even though an acute reader noted that a piece of the program was in the wrong place, and where it should have been, and the designer of the machine confirmed the fix. – supercat Mar 14 at 21:56

@supercat: Danny Hillis and Brian Silverman built two editions of the TinkerToy tic-tac-toe player. - Version 1, in 1978, uses three-state logic gates, and is a cube about one meter on a side. This went first to the Mid-America Museum of the Arkansas Department of Parks and Tourism, and then to the Computer Museum in Boston, where it was on display in 1989. - Version 2, in 1980, uses serial logic, and looks like a demented clockwork wall tapestry. It was built because the Mid-America Museum wanted a more robust version. This is the one described in the October 1989 Scientific American. – A. I. Breveleri Mar 14 at 23:52

4

A 1949 film clip hints at an electromechanical implementaton. The information content is low, but a picture of a uniselector suggest it's switches-and-relays logic.

https://www.britishpathe.com/video/noughts-and-crosses-machine

1949 was definitely in the stored-program computer age (EDSAC ran its first program in May 1949) but this clearly was not computer technology.

Share Improve this answer Follow

answered Mar 14 at 4:22

another-dave

**16.6k** 4 31 81

3

## With only wires and lights: NO.

The straight answer is NO. It can not be done, as neither wires not 'light globes' are active elements, thus they can't be modified/rearranged to model the game. Any solution will at least need a

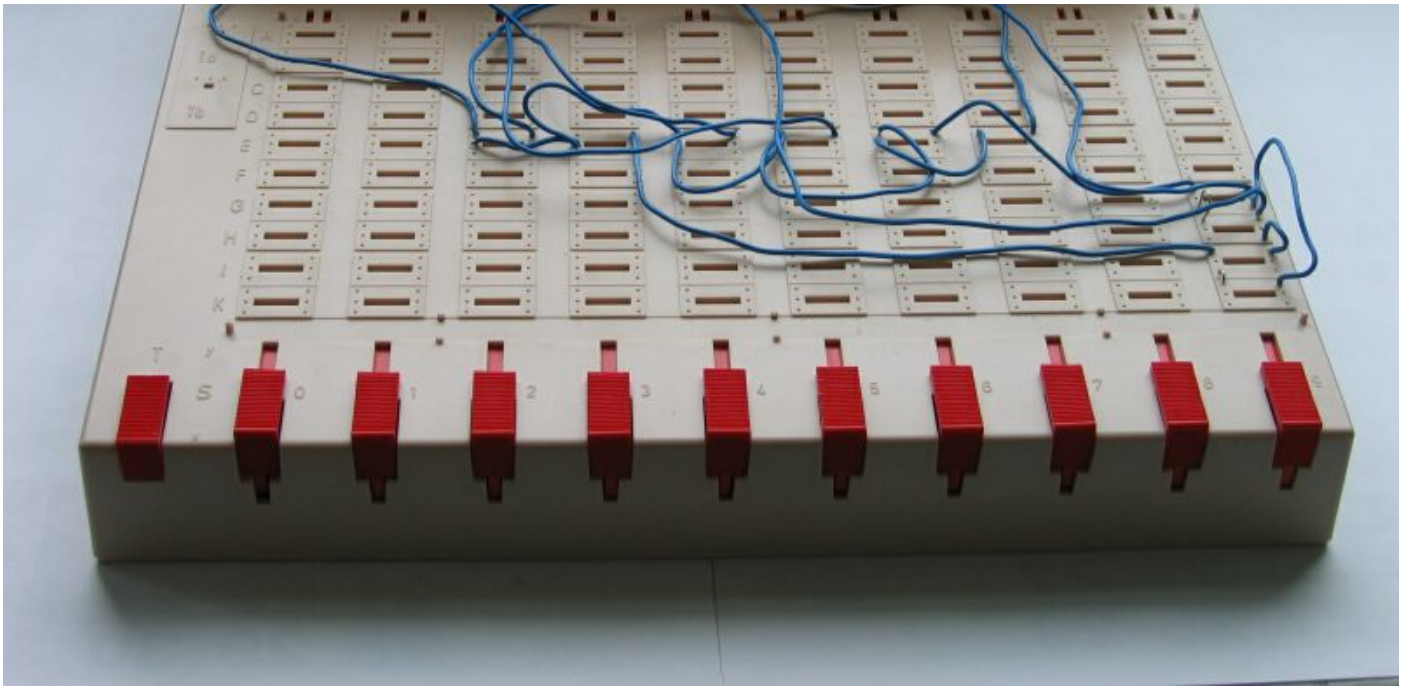
combination of complex (read multi contact) switches that will be moved either manually or machine controlled (which in addition needs clocking, like due a motor).

## But YES if we add switches

Noughts-and-crosses 'programming' follows an extreme simple and straight forward scheme, allowing to draw up a fixed tree of moves. As a result this can be implemented as a complex but terminal number of switches and connections inbetween.

## The Logikus

An early device (and general available) that possible could be used to 'encode this' would be the 1968 Logicus by Kosmos:



*(picture taken from here)*

There is a German Wiki-entry, as well as a whole site dedicated to the Logikus. A bit more information (in English) can be gathered from a page about an emulator.

There were at least two **licenced versions in the US and Canada** by Logix Enterprises of Montreal and New York called O-600 and SF-5000, ca 1970, as well as kind of a simpler clone by Tandy as part of their "Science Fair" Series (as late as 1977). There's an English language video describing it - and another, showing more details, although she doesn't seem as exited :))

For this I's stay with the original.

## Workings

It was an educational toy, first produced in 1968, consisting of 10 switches, 10 light bulbs and a button to power whatever is connected when pressed.

- Each switch could be in one of two positions, up or down.
- Each switch had 10 contacts, of them
  - half were closed
  - and half were open in either position
- Each contact could be feed by up to three wires
- Each contact could feed 3 wires.

Each light bulb could be feed by a single wire.

The button was connected to the batteries and all light bulbs.

To plug a 'program' wires would connect the button with some contacts of some switches and routed from there. Connecting two contacts

- in series forms an AND relation
- in parallel forms an OR relation

Doing either over 'open' contacts negates logic, introducing NOT

In combination any switch can be seen as a single bit input, representing two possible values, while each lamp was a single bit output as lit or dark.

## Implementing the game

While a single Logikus has not enough switches and contacts to represent the game in a plain version, using prepared input (which in turn could come from one or more other Logikus') could allow to program the strategy on a single Logikus:

*(just a quick write up, I bet I forgot some combination)*

Lets have binary inputs for

- Two own tokens and an empty field in a row
- No token at center
- One own token plus two empty in a row
- No token in corner

These 4 conditions can easy be combined to 4 output bits to lit one of 4 commands to the player:

- Place token in empty field and win
- Place token at center
- Place token in row with one own and two empty (in corner)
- Place token in empty corner

All of these can be made with easy AND/OR/NOT connections.

For a detailed output which field is to be used this result can be used in conjunction with the input data (from previous stage) to form an output stage naming exactly which field the token is to be place usind another Logikus.

## Conclusion

While this can be used to play the whole game - sufficient switches/contacts given - it would not play it automatic, as there are no sensors, actors and stepping. All it can do is to value a certain game state and derivate the next move. Everything else, from inputting the state to executing the move must be done by the operator.

**As mentioned, due usage of switches this already violates the boundaries set by the question.**

Share Improve this answer Follow

edited Mar 16 at 14:38

answered Mar 14 at 5:15

Raffzahn

**130k** 16 335 523

Just an insane commentary but wires can (and are) used as active non linear parts ... For example it can be used as Diac you just need high voltage. Light bulbs + with wires can be used as diodes however that would be extremly ineffective if the missing electrode is not inside the light bulb... However who would even try to touch such "game" device ... as it would most likely leak some RTG and a lot of UV and Ozone not to mention the burning and other high voltage High Current health issues – Spektre Mar 14 at 9:14

@Spektre Sounds like the perfect property for a 1940s Frankenstein film – Raffzahn Mar 14 at 20:24

I wonder how many switches would be required to have a machine which, if switches were set for all positions where players had put X's, would illuminate lights for all positions where the machine would have placed O's. If one restricts the player's first move to square 1, 2, or 5, that would limit the number of combinations of cells that a player could claim. – supercat Mar 14 at 21:47

@supercat No many. I say 9 three way switches. One for each field (position), with three possible settings (empty, X, O). The real issue will be the number of contacts per switch, as they 'encode' the logic equations. Their number depends on how much the equations can be simplified (reduced), but will be rather high without intermediate (memory and multiplexer) elements. MENACE does prove that a static solution is available, which is the most important part here. As more I think about as more it seems doable. even with minimal hardware - but not without switches/contacts. – Raffzahn Mar 14 at 21:57

@Raffzahn: I meant how many contacts. I think one could get by with two-position switches if all squares played by the human remained played (if e.g. the human went first and has played on 2 and 5, then assuming the human hasn't cheated the computer should have squares 1 and 8). – supercat Mar 14 at 22:28

@s it needs to have a third state of 'computer has played'. Otherwise the same move over and over

would be presented as the next to be done. To calculate the contacts the number of equations and all operations/operand combinations within need to be known, as theareerepresented bei a contact (either open or close) on one of the switches. – Raffzahn Mar 14 at 22:33

@Raffzahn: If the input is an unordered set of all the moves the human has played thus far, and optimal play isn't required, I don't think state is required. If e.g. the game's response to an opening 5 is 1 and the player has occupied 5 but not 9, then for the human to occupy 2 should light 8, 3 lights 7, 4 lights 6, 6 lights 4, 7 lights 2, and 8 lights 1. If the human has played 5 and then plays 6 and 8 in some order, the optimal set of moves for the computer to have made would depend upon the order of the last two human moves, but the computer could legally light 1, 2, and 4. – supercat Mar 15 at 0:23

@Raffzahn: Note that if the human has made any moves but hasn't claimed 5, then the computer should claim 5, so the only way switch 5 can be closed is if that was the first move. I'm not sure if legal play for center or edge openings could be accomplished without any state other than an unordered set of human moves, but for center openings I think legal but not optimal play (non-losing, but not pursuing chances to win against mistakes) can be done statelessly. – supercat Mar 15 at 0:26

@supercat Beside that there is no way to do an 'unordered set' without memory an loops, and that such an unordered set would as well be a state, there is no need to do so, as the games data set is static and complete. So 9 switches of 3 positions ill code all input there ever might be.- likewise 9 lamps will be enough to call the next move. In fact, the 'machine' can play against itself with the same setup. No need to make any setup or computation more complicated then necessary. – Raffzahn Mar 15 at 0:28

@Raffzahn: An unordered set is simply a collection of switches. A switch is on if an opponent has made a move to that square, and left on while making subsequent moves. – supercat Mar 15 at 0:29

@Raffzahn OMG I had a Tandy (Radioshack) version of that I bought in the 70s with my (limited) pocket money because the box had a picture of it playing chess. Talk about misleading packaging!! – Simon F Mar 16 at 14:12

@supercat A switch can't be unordered. – Raffzahn Mar 16 at 14:14

@SimonF There was one by Tandy? So far I only knew about the Logix clone in the US. Googling brought up a video. Cool. I need to add that. thanks. – Raffzahn Mar 16 at 14:35

@Raffzahn: A set of nine switches is capable of representing the 512 unordered sets of the digits 1-9, including the empty set (all switches off) and the all-members-present set (all switches on). If switches 5, 6, and 8 have been switched on, and the others haven't yet, then the switches will be in the configuration off-off-off-off-on-on-off-on-off. – supercat Mar 16 at 15:24

@Raffzahn: If the machine is in the state where just 5 is on, that state will turn on light 1. If 5 and 8 are on, then lights 1 and 2 should be on. If 6 is then turned on, optimal play would have lights 1, 2, and 3 on, but having lights 1, 2, and 4 on would be a valid and non-losing play regardless of whether the player flipped the switches in the order 5-8-6 or 5-6-8. Note that if the player had operated switch 8 before switch 5, the machine would have lit switch 5, and so for the player to turn on switch 5 at that point would have been cheating, so the fact that... – supercat Mar 16 at 15:28

...lights 1 and 2 would turn on simultaneously in response to that wouldn't be a defect. For optimal play on a device which has no switching of its own, it would be necessary to add at least one more light and switch, with instructions that if an extra light goes on, the player must flip the extra switch. Then if the switches are in state 5-8 the machine could light 1-2 and if in state 5-6 it could light 1-4-X. If switches are in state 5-6-8 the machine could light 1-2-3 and if in state 5-6-8-X it could light 1-4-7. – supercat Mar 16 at 15:31

@Raffzahn Saw the video. Strange, that's different to what I had... maybe they had cost-reduced it by the time I bought mine. For example, the one in that video had springs to connect the wires. I'm sure mine only had holes you poked the wires into, a little bit like a breadboard... but cheaper and nastier! – Simon F Mar 17 at 14:04

@Raffzahn Found one! radioshackcatalogs.com/flipbook/1976_radioshack_catalog.html Go to page 94 bottom right corner! – Simon F Mar 17 at 14:18

@SimonF Jup, that was a 1:1 Logikus/Logix clone, a tactic Tandy often applied, but here they got sued by Logix, NY, and had to come up with a different design, which was the one with the springs like all other Science Fair units. – Raffzahn Mar 17 at 14:47